Distributed Identity on the Web: Usability and Safety

Sam Hartman

Massachusetts Institute of Technology

June 15, 2006

The Web is Bleak

- Today’s web single-sign-on solutions focus on avoiding browser changes.
- Passwords are sent to login servers.
- Login servers become large targets for phishing.

What if we could change the browsers?

Use Case: Alice Uses the Web

- Alice selects login and chooses an identity.

- A trusted local UI prompts for passwords if needed.

- Alice needs a new identity only when privacy or policy dictates—as close to single-sign-on as possible.

Use Case: The Web is Safe for Alice

- Alice’s password is never sent to the server; her password cannot be phished.

- Alice can reuse passwords between sites with relative security.

- Alice knows the web page she eventually received is from the party she authenticated to.

- Enrollment of new accounts does not disclose Alice’s password.

Use Case: Bob Deploys Distributed Identity

- Bob decides which identities he accepts.

- Accepting multiple identities is easier for Alice but may not meet Bob’s business constraints.

- Bob upgrades his servers and adds simple controls to his HTML.

We can change the browsers!

Phishing: Safety Alice can Understand

- Alice brands her computer when she installs it.
- She sees this branding only when entering passwords into a trusted UI; no branding when passwords would be sent over the net.
- Attackers don’t know Alice’s branding so they cannot fake it.

Phishing: Safety Alice can Understand (2)

- Once Alice has safely authenticated the website can prove its identity by displaying confidential information only Alice and the website know.

- Examples: bank transactions, recent orders, website specific branding Alice chose

- Works only with existing relationships.

- This is not safe on the web today: Alice's password can be phished.

Phishing: Requirements to make it Work

- Password equivalents never sent across the net
- Mutual authentication of the server
- Binding of the returned web page to the authentication
- Support for identities accepted by a small number of servers

What should the browsers do?

Solution: Overview

- HTML extensions describe the website’s identity requirements.
- HTTP Negotiate authentication carries identity information.
- Kerberos provides distributed identity.
- SAML describes claims about the identity.

Solution: Negotiate Authentication

- Introduced with Windows 2000 for enterprise web authentication; widely deployed

- Good user experience once enabled

- Needs to be standardized and moved beyond a Microsoft protocol

Solution: Usage of Kerberos

- Servers get a principal in the realm of each identity provider they accept; cross realm can be used but assuming cross-realm relationships limits deployability.

- A firewall-friendly transport is needed.

- Automated enrollment of new identities and servers into a realm is needed.

- For Internet deployment, enrollment can be bootstrapped from TLS certificates or leap-of-faith.

How can **you** change the browsers?

Additional Reading

- `http://tools.ietf.org/id/draft-hartman-webauth`
- `http://tools.ietf.org/id/draft-hartman-webauth-phishing`

How you Should Get Involved

Join `dix@ietf.org`, contribute to the discussion and help people understand this is a useful problem to solve.

Review proposals and send comments.

Attend IETF in Montreal or listen to the audio stream and contribute via Jabber.

IETf Information

- Meeting July 9 through 14th
- `ietf.org` for meeting agenda and for timing of the discussion
- Look for WAE BOF